SAP Solution in Detail

MaxDB™: THE PROFESSIONAL DATABASE FOR TODAY'S ENTERPRISE ENVIRONMENT

© Copyright 2005 SAP AG. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP AG. The information contained herein may be changed without prior notice.

Some software products marketed by SAP AG and its distributors contain proprietary software components of other software vendors.

Microsoft, Windows, Outlook, and PowerPoint are registered trademarks of Microsoft Corporation.

IBM, DB2, DB2 Universal Database, OS/2, Parallel Sysplex, MVS/ESA, AIX, S/390, AS/400, OS/390, OS/400, iSeries, pSeries, xSeries, zSeries, z/OS, AFP, Intelligent Miner, WebSphere, Netfinity, Tivoli, and Informix are trademarks or registered trademarks of IBM Corporation in the United States and/or other countries.

Oracle is a registered trademark of Oracle Corporation.

UNIX, X/Open, OSF/1, and Motif are registered trademarks of the Open Group.

Citrix, ICA, Program Neighborhood, MetaFrame, WinFrame, VideoFrame, and MultiWin are trademarks or registered trademarks of Citrix Systems, Inc.

HTML, XML, XHTML and W3C are trademarks or registered trademarks of W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

Java is a registered trademark of Sun Microsystems, Inc.

JavaScript is a registered trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

MaxDB is a trademark of MySQL AB, Sweden.

SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies. Data contained in this document serves informational purposes only. National product specifications may vary.

These materials are subject to change without notice. These materials are provided by SAP AG and its affiliated companies ("SAP Group") for informational purposes only, without representation or warranty of any kind, and SAP Group shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP Group products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

CONTENTS

What is MaxDB? **4**

Meeting the Challenges of an Enterprise Environment **5**
How SAP DB Became MaxDB 5
Component Portfolio 6
Operating System Platforms Supported by MaxDB 6

Features and Functions of MaxDB **6**
Data Organization and Storage 6
Transactional Control of Data Changes 7
Data Security and Consistency 7
High Availability 9
Performance 10
Control of User Access to Data 11
Secure Communication 12

Using the Database Management System **12**
Managing Database Instances 12
Defining and Changing Data Using SQL 13
Monitoring the Database and Analyzing Problems 13
Importing and Exporting Data 14
Replicating Data 14

Developing Database Applications **15**
Programming Interfaces 15

Implementing MaxDB **16**
Download 16
Installation 16
Licensing Model 17
Consulting 17
Support 17
Training and Certification 18
– SAP Training 18
– Training by MySQL 18
– Certification by SAP 19
– Certification by MySQL 19
– Technical Specifications 19
MaxDB Documentation Overview 19

Why You Need MaxDB **21**
For More Information 21

WHAT IS MaxDB?

MaxDB™ is the enterprise database management system (DBMS) you need to run your company efficiently. Easy to use, it offers low cost of ownership, outstanding performance, and much more. MaxDB supports a broad range of platforms, a high number of users, and demanding workloads. It is highly scalable, available, and reliable. Administration costs are low because there are not many administrative tasks to execute. You can do most of them while the database is online. This online approach, which eliminates the need for costly maintenance downtime, is a prerequisite for adaptive computing and self-management. It is also the reason MaxDB is less expensive than other enterprise DBMSs.

MaxDB is a rebranded and enhanced version of the SAP® open source database formerly called SAP DB. (See sidebar.) It has been used for many years to support small, medium, and large installations of the mySAP™ Business Suite family of business solutions. It has also been used for demanding SQL applications requiring an enterprise-class DBMS because of the number of users on the system, the transactional workload, and the size of the database.

Today, MaxDB is being used in about 3,500 SAP customer installations worldwide. In addition, MaxDB technology is bundled with SAP solutions, such as mySAP Supply Chain Management, SAP Solution Manager, and multisite workforce deployment (WFD), and so on, in approximately 7,000 customer installations. It is the database of choice for the majority of internal SAP installations on UNIX and Linux. MaxDB is designed for heavy-duty online transaction processing (OLTP) with several thousand users and supports database sizes ranging from several hundred gigabytes (GB) to multiple terabytes (TB).

Despite its suitability as an enterprise-class DBMS, MaxDB is not restricted to multiuser, server-based situations. Because its ease-of-use capabilities are built in, not built around, it can also be used as a desktop or laptop DBMS. In these environments, its self-managing capabilities turn it into an "invisible" DBMS.

Commercial licenses and support for MaxDB are available from SAP and MySQL. For open source or other software development projects and for trial or educational purposes, MaxDB is also available as open source under the GNU General Public License (GPL), the most popular open source license. To develop commercial software based on MaxDB, you need a commercial MaxDB version that enables you to deploy the software at your customer's site.

To implement MaxDB, you can either use one of the compiled "executables" available for prominent operating system platforms or compile your own executable programs using the MaxDB source code. Downloads of the MaxDB development environment and released versions of the MaxDB sources are available at **www.mysql.com/maxdb**. You can also use Concurrent Versions System (CVS) to download the latest MaxDB source code.

MEETING THE CHALLENGES OF AN ENTERPRISE ENVIRONMENT

MaxDB provides the solutions and tools you need to respond effectively to the challenges you encounter in today's enterprise environment. It offers:

- **Proven DBMS technology** – based on many years of experience with OLTP, data warehousing, decision support, and object-oriented (SAP liveCache) applications
- **Attractive pricing**
- **Easy configuration and administration** – achieved by incorporating a GUI-based MaxDB installation manager and MaxDB database manager as single administration tools for DBMS operations
- **Around-the-clock operation, no planned downtime** – thanks to automatic space management, which ensures that permanent attendance and reorganizations are not required
- **Elaborate backup and restore capabilities** – including online and incremental backups as well as a wizard to guide you through the recovery process
- **Support for a large number of users and demanding workloads** – ensuring proven reliability, performance, and scalability for terabyte-size databases
- **High availability** – thanks to cluster support, standby configuration, and hot standby configuration
- **Tools** – including MaxDB installation manager, MaxDB database manager, MaxDB SQL studio, MaxDB loader, MaxDB synchronization manager to replicate data between MaxDB instances, and a WebDAV-based document and XML repository
- **Interfaces** – including ODBC, JDBC, SQLDBC, PHP, Perl, and Python

Most important, MaxDB:

- Is available for all major operating system platforms
- Supports all major SAP solutions
- Is bundled with the SAP NetWeaver™ platform

How SAP® DB Became MaxDB™

In 2003 SAP and MySQL concluded a partnership and development cooperation agreement. As a result, the SAP® open source database, SAP DB, is being delivered under the name MaxDB – and has been since the release of version 7.5.

Version 7.5 of MaxDB is a direct advancement of the SAP DB 7.4 code basis. Therefore, the MaxDB software version 7.5 can be used as a direct upgrade of previous SAP DB versions starting 7.2.04 and higher.

The former SAP DB development team at SAP AG is now responsible for developing and supporting MaxDB. Sales and distribution are handled by both SAP AG and MySQL AB. The advancement of MaxDB and the MySQL server leverages synergies that benefit both product lines.

MaxDB is subjected to the SAP AG quality assurance process before it is shipped with SAP solutions or provided as a download from the MySQL site.

Unlike other SQL DBMSs, MaxDB offers patented SAP liveCache technology, an object-oriented DBMS extension for C++ applications. When used for large, tree-type or network-type data models, this technology improves performance by a factor of 20 to 50 compared with OLTP DBMSs. SAP liveCache technology and applications (written as stored procedures in C++) are the cornerstone of the mySAP Supply Chain Management (SCM) solution. mySAP SCM customers can choose an SCM One DB configuration that combines MaxDB and SAP liveCache in a single database to reduce the need for database administration (thus reducing cost of ownership).

Component Portfolio

The MaxDB portfolio includes tools for installation and administration, tools for querying and moving data, and interfaces for database programming. (See Figure 1.)

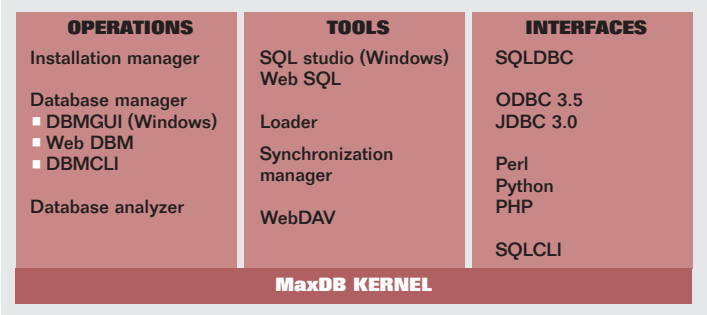


Figure 1: MaxDB Component Portfolio

Operating System Platforms Supported by MaxDB

MaxDB is available on the following operating systems and hardware processor architectures:

Operating System	Hardware Architecture
HP-UX	HP-Risc IA64
IBM AIX	Power
Linux	X86_32 X86_64 IA64 Power
SUN Solaris	SPARC
Microsoft Windows 2000	X86_32
Microsoft Windows XP	X86_32
Microsoft Windows Server 2003	X86_32 X86_64 IA64

FEATURES AND FUNCTIONS OF MaxDB

MaxDB simplifies enterprise data management tasks through a wide range of features and functions, detailed in this section.

Data Organization and Storage

Application data, including metadata and a log of the latest data changes, is contained in the **database instance**. The application and log data are kept in volumes. Volumes can be complete physical disks, parts of a physical disk, or files in a file system. Each database instance has at least one log and one data volume. Volumes contain a fixed number of 8-kilobyte (KB) pages and are thus a fixed size. The growth of a database or the demand for more log space is managed by adding further data or log volumes to its configuration.

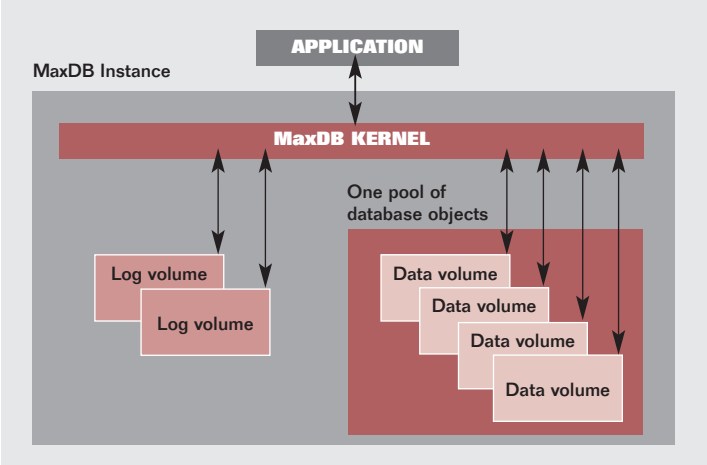


Figure 2: MaxDB Database Instance

The total size of a MaxDB instance is limited by the sum of all data volume sizes. MaxDB does not require any size estimations for individual database objects (such as tables and indexes). Each table can grow and shrink without any administrative action (automatic space management). All data of all tables is distributed on all data volumes. Therefore, tables cannot be assigned to any particular data volume, and vice versa.

The implicit distribution of all data offers several advantages. The growth is distributed equally among all data volumes and the input/output (I/O) load is spread evenly among all data volumes. Consequently, there is no need and no possibility for the database administrator (DBA) to fine-tune table or index allocations to any particular volume in order to balance the I/O load because this is done automatically by MaxDB. When MaxDB is close to full, it can be extended by additional volumes while it is online. Such data volume extensions can be carried out manually by the DBA or automatically by MaxDB. Moreover, an automatic procedure can be used to back up the database log, preventing the log volumes from filling up.

SQL DBMSs generally provide high abstractions with respect to the physical aspects of data management. MaxDB takes this one step further by defining all database objects (for example, tables) on a purely logical level. No DBA assistance is required in defining the properties of their physical storage. For example, it is indeed possible to obtain information about the size and internal structure of a table or an index, but there is no requirement and therefore no possibility to influence the physical storage by means of any low-level directives. This reduces both administration efforts and training costs.

Transactional Control of Data Changes

Database applications carry out changes of data under the control of transactions, which consist of consecutive SQL statements that are treated as a unit by the database system. Consequently, unless **all** changes of a transaction are executed, the transaction will have no effect. In a standard transaction – for example, rebooking the amount of $100 from account A to account B – the sum of $100 either has to be debited from account A and credited to account B, or it has to remain available on account A. It is impossible to have an error situation in which the $100 is missing from account A but has not yet been credited to account B.

Transaction management ensures that all transactions for all concurrent DBMS users are processed and that the database is in a defined, consistent, and restorable condition at any time and for every possible error situation (for example, a power failure).

MaxDB is never in "transactionless" mode. The user login implicitly triggers the first transaction. Once the transaction has been successfully terminated by a "commit," all changes are made permanent. If the transaction is ended by a "rollback" or is otherwise terminated, all data changes made within the transaction are rolled back. Both "commit" and "rollback" will terminate the transaction and implicitly open a new transaction. For interactive query tools (MaxDB SQL studio, MaxDB Web SQL), an "autocommit" option implicitly turns each single command into a separate transaction.

Data Security and Consistency

MaxDB stores all data-object change operations in log volumes. When the database is restarted after an unplanned downtime, it uses these log entries to decide on completed and uncompleted transactions and to restore the latest consistent state of the database.

The **transaction log** ensures transactional consistency but will not protect from media errors – that is, data loss. This is why you have to have a backup strategy in place for each productive MaxDB system – a strategy based on periodic data and log backups.

You can create the **backup strategy** that suits you best. For example, you can perform a complete data backup (or at least an incremental backup) on each productive day to back up all changes made since the last complete data backup. The more recent the previous backup, the fewer log entries will have to be reworked to restore the database instance.

Or you can initiate an incremental backup strategy that is limited to the pages that were actually changed since the last complete backup. Whichever method you choose, you must follow it faithfully. Log backups are required; they record the change history of a database upon the latest data volume backup.

MaxDB allows for data and log backup during operation with little noticeable effect on overall performance. To accelerate the backup of very large database instances, write the backup

to multiple media (tapes, disks) in parallel data streams. Such parallelization is also possible when restoring a database instance to minimize potential downtime.

MaxDB also provides an automatic log backup (an "autosave" log) that should be switched on at all times. This enables the DBMS to automatically back up the log volumes to a set of files once the log reaches a precalculated filling level.

The MaxDB database manager logs all backup activities and assists you during the recovery process. The recovery wizard first offers the most likely recovery path. From there, you can control the recovery process by selecting the alternatives that will work best for you.

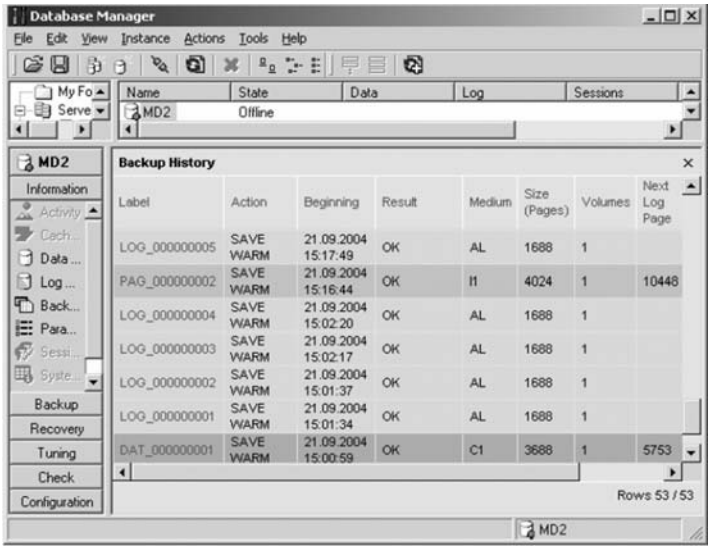

Database Manager
File Edit View Instance Actions Tools Help
Name State Data Log Sessions
MD2 Offline
MD2
Backup History
Label Action Beginning Result Medium Size (Pages) Volumes Next Log Page
LOG_000000005 SAVE WARM 21.09.2004 15:17:49 OK AL 1688 1
PAG_000000002 SAVE WARM 21.09.2004 15:16:44 OK I1 4024 1 10448
LOG_000000004 SAVE WARM 21.09.2004 15:02:20 OK AL 1688 1
LOG_000000003 SAVE WARM 21.09.2004 15:02:17 OK AL 1688 1
LOG_000000002 SAVE WARM 21.09.2004 15:01:37 OK AL 1688 1
LOG_000000001 SAVE WARM 21.09.2004 15:01:34 OK AL 1688 1
DAT_000000001 SAVE WARM 21.09.2004 15:00:59 OK C1 3688 1 5753
Rows 53 / 53
Information
Backup
Recovery
Tuning
Check
Configuration
MD2

Figure 3: Backup History

The screen shown in Figure 3 is an extract from a backup history at the start of a backup generation. It shows a complete backup (DAT_. . .), several log backups (LOG_. . .), and an incremental backup (PAG_. . .). The MaxDB database manager uses this log to select the appropriate data backups for the **recovery process** shown in Figure 4.

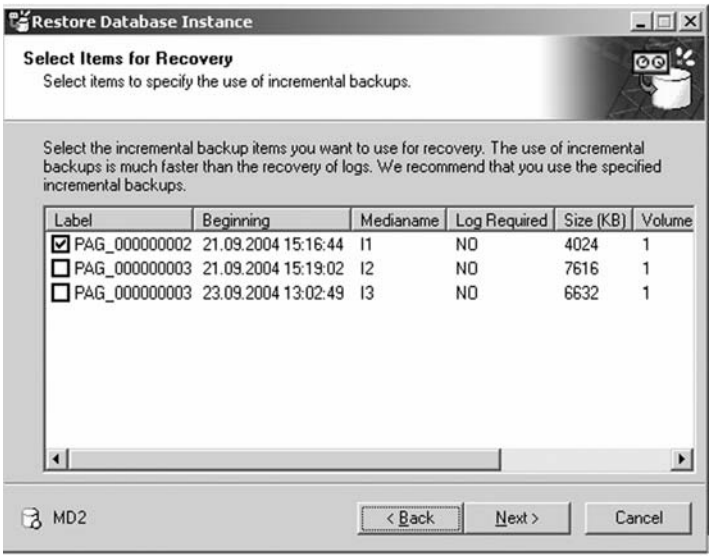

Restore Database Instance
Select Items for Recovery
Select items to specify the use of incremental backups.
Select the incremental backup items you want to use for recovery. The use of incremental backups is much faster than the recovery of logs. We recommend that you use the specified incremental backups.
Label Beginning Medianame Log Required Size (KB) Volume
PAG_000000002 21.09.2004 15:16:44 I1 NO 4024 1
PAG_000000003 21.09.2004 15:19:02 I2 NO 7616 1
PAG_000000003 23.09.2004 13:02:49 I3 NO 6632 1
MD2
< Back
Next >
Cancel

Figure 4: Select Recovery Path

In Figure 5, the incremental backup results in the selected recovery path suggested by the recovery wizard.

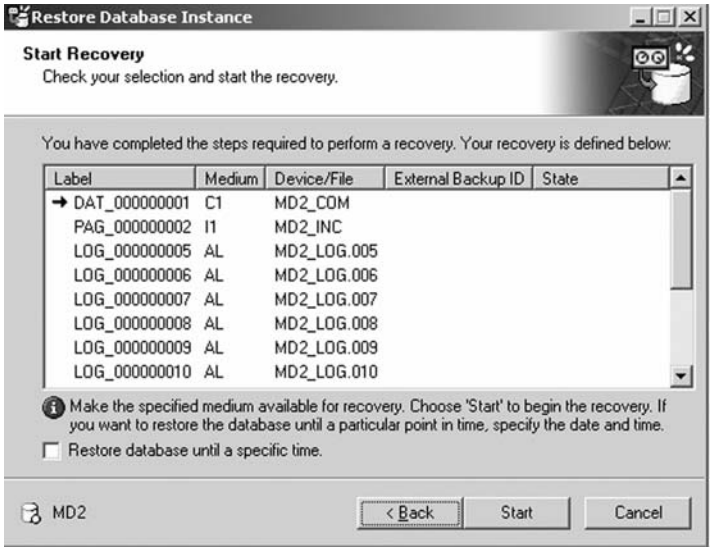

Restore Database Instance
Start Recovery
Check your selection and start the recovery.
You have completed the steps required to perform a recovery. Your recovery is defined below:
Label Medium Device/File External Backup ID State
DAT_000000001 C1 MD2_COM
PAG_000000002 I1 MD2_INC
LOG_000000005 AL MD2_LOG.005
LOG_000000006 AL MD2_LOG.006
LOG_000000007 AL MD2_LOG.007
LOG_000000008 AL MD2_LOG.008
LOG_000000009 AL MD2_LOG.009
LOG_000000010 AL MD2_LOG.010
Make the specified medium available for recovery. Choose 'Start' to begin the recovery. If you want to restore the database until a particular point in time, specify the date and time.
Restore database until a specific time.
MD2
< Back
Start
Cancel

Figure 5: Recovery Path Display

In the event of media problems – for example, a defective data volume – the DBA has to recover the entire database once the media failure has been repaired. The recovery is effected by importing the latest database backup to the data volumes and redoing the latest log backups.

Recovery can also be used to restore a previous database state (point-in-time recovery). This may be required if application problems or administration errors occur (for example, if a table is accidentally deleted).

High Availability

Database instances can fail or become defective as a result of hardware errors such as disk or controller defects, power failure, and operating system failures – or as a result of operating errors (for example, tables unintentionally deleted or incorrect data imported or processed).

To avoid downtime in these situations, MaxDB offers several defined procedures to support high-mission-critical database instances. These procedures include the options of standby or hot standby configurations.

A **standby database** is a copy of the active database instance (master). Once it has been initialized on the basis of a data backup of the master instance, the log backups of the master are continuously imported into the standby. As a result, all changes to the master are replicated in the standby. Consequently, the standby instance always shows the data set of the master instance per the most recent log backup imported.

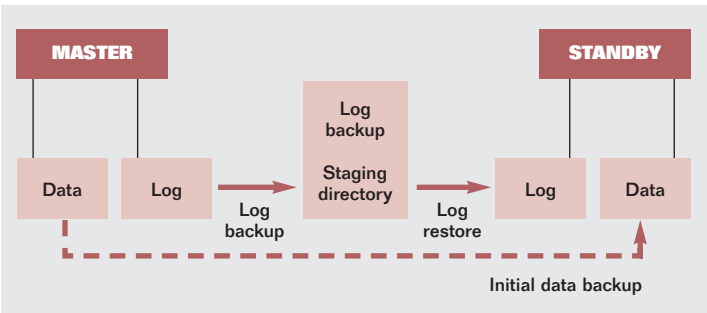


Figure 6: Standby Database with Log Shipping

In the event of problems with the master, a standby can be put into operation immediately, and work can be continued without a lengthy recovery process. Simply make sure that the remaining log from the master instance is transferred to the standby instance.

This configuration is supported by third-party tools such as Libelle (**www.libelle.de**).

Hot standby refers to the MaxDB high-availability strategy, which allows you to switch over to a parallel standby database instance within seconds should the active database instance fail. A hot standby system consists of an active master instance and one or several standby components installed in an operating system cluster configuration.

To effect the hot standby strategy, you need an intelligent storage system with direct copy facilities (BCV, FlashCopy, Snapshot). The standby component is initialized by a direct copy within the storage system. Once the copy has been made, the standby component reads in the master log entries, which detail which transactions the master component has executed. Then it writes the resulting database changes to the standby data volumes.

Outwardly, a hot standby configuration behaves like a single database instance. Figure 7 shows how the configuration is structured. In the event of a master failure, the IP routing is redirected to the standby node, which restarts immediately and is available to the application again within seconds, preventing any loss of committed data. Thus the standby automatically becomes the master. Once the defect on the former master has been eliminated, that former master can be initialized for standby usage, completing the configuration. Switching from master to standby cancels the transactions (rollback) on the various current applications.

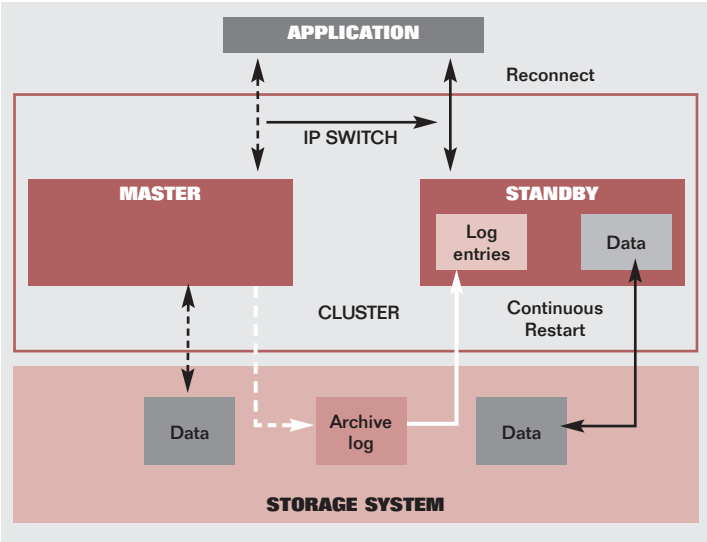


Figure 7: Hot Standby Switchover Situation

You can also define a time interval by which the standby is to lag behind the master. This supplies a time window for correcting logical errors in database operations, which can be caused by the DBA, incorrect data, or incorrect applications.

In addition to the high-availability options described earlier, MaxDB also lets you create a consistent **snapshot of a complete database state**. The current database state is "frozen" and can be restored again if required.

This snapshot process can be done quickly, since no copying of data is involved. It prevents logical database errors and is thus suitable for establishing fallbacks (for example, during application upgrades). It can also be used to reset training systems to their initial state.

Because no additional backup media are required for a snapshot, it will not protect you from media loss.

Performance

MaxDB is based on a **multiprocess/multi-threaded server architecture** geared toward optimal scalability on symmetric multiprocessor systems. Each database instance uses several operating system processes and implements internal threading.

MaxDB does not rely on operating system threads but on internal threads that release database resources (such as locks on "commit") that can be prioritized, increasing the database throughput. Internal threading also enables the DBMS to distribute the workload optimally.

The read and write operations on the data of a database instance are buffered in order to minimize physical disk access. These buffers or **caches** can be configured using database parameters. Major MaxDB caches include the data cache and the catalog cache for the SQL catalog (metadata).

MaxDB uses a special variant of prefix B* trees for its disk storage organization. There is a B* tree for each table (primary data) as well as a B* tree for each index of a table (secondary data).

Automatic space management and automatic balancing of the I/O workload across all data volumes provide disk storage organization that does not require any reorganization whatsoever. It will not degenerate even after years of high-volume write operations.

All data structures are checked for internal consistency. As a result, potential hardware defects are detected and isolated as soon as possible.

MaxDB uses a **cost-based query optimizer** to determine the best execution plan for a "select" command. Cost-based optimizers typically use statistical data about table, index, and column cardinalities as well as value distributions.

But outdated statistical data can cause the optimizer to make a wrong decision (calculation) on how to best access data. For single table queries, the cost-based optimization of MaxDB is not dependent on statistical data. Instead it uses actual cardinalities that are administered within the scope of the B* tree organization, thus eliminating the possibility of incorrect optimizations caused by outdated statistical data. The process of optimizing "joins" requires up-to-date statistics. MaxDB detects when these statistics are no longer in sync with the actual database cardinalities and places a "refresh" request with the DBA.

Alternative query plans are created during preparation of a "select" command. The final query execution strategy is determined at execution time once all actual parameter values are known. Additionally, query rewrites are applied to optimize the execution speed of certain "select" commands.

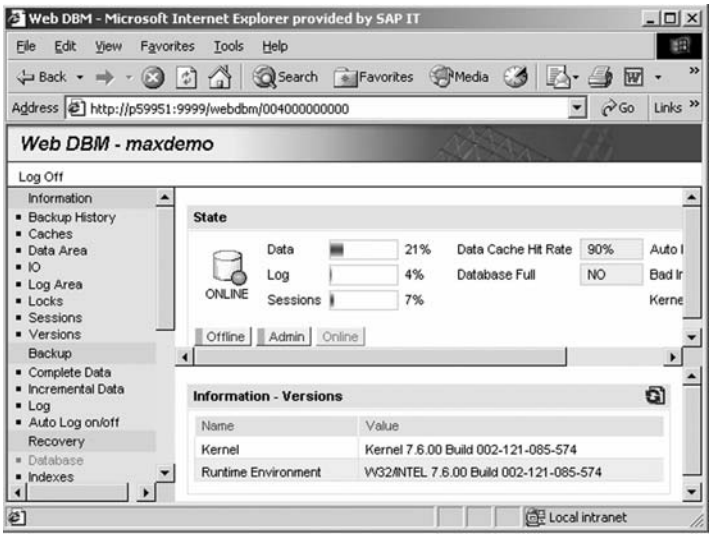


Figure 8: Browser-Based Administration (MaxDB Web Database Manager)

Control of User Access to Data

MaxDB distinguishes two types of users: **database users** and **database administrators** (users of the MaxDB data manager tool). Database users work with the database objects using SQL, while DBM users monitor and service their DBMS instance(s) using the administration tool.

Every object in the database – tables, indexes, views, procedures, users, authorizations, synonyms, sequences – is described and managed in the SQL catalog. Special authorizations for catalog administration can be assigned to specific users.

MaxDB database administrators have no permission to access the database content. They have no SQL access to the database nor do they have permission to create new database user accounts. MaxDB database manager merely permits them to manage and monitor the database instances.

The creator of a data object implicitly becomes the owner of the object he or she creates and has all the authorizations (privileges) for it. The owner can then forward access privileges ("select," "update," "delete," "insert," and more) to other database users.

A default **schema** is created implicitly "behind" the name of every user entered in the system. The schema groups the data objects associated with that person into a separate namespace to avoid naming conflicts. (Schemas are analogous to folders in a file system.) Additional schemas can be created explicitly and filled with new database objects.

In Figure 9, the user Mona has explicitly created the additional schema "4You" to her default schema, and has assigned the rights to create objects in this schema to the users Bob and Carl. These users then added their own tables to this schema.

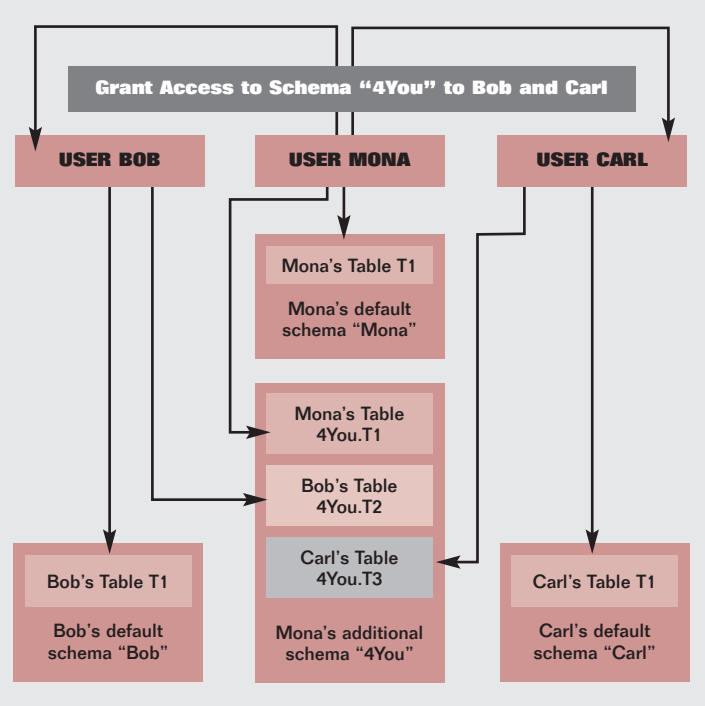


Figure 9: Schemas

Schemas are used to decouple the database users from name spaces. With schemas, the user name is no longer part of the name of a database object, which enhances DBMS flexibility.

Secure Communication

As an option, the socket communication between MaxDB client and MaxDB server can be safeguarded by means of SSL. This is a security requirement for certain Internet scenarios. If the database and its clients are implemented in a secure in-house environment, SSL support can be deactivated.

The MaxDB authentication procedure is a challenge-response procedure so that passwords that may have been captured cannot be used to open any other session.

USING THE DATABASE MANAGEMENT SYSTEM

MaxDB provides a variety of tools for working with database instances. All MaxDB tools can be used in connection with both SAP solutions and other applications.

Managing Database Instances

The **MaxDB database manager** is a tool that permits you to operate an entire landscape of MaxDB instances centrally. It can create and monitor the instances on local or remote computers, back up the data on them, and restore it if required.

MaxDB database manager is available as a Windows-based tool **(MaxDB database manager GUI)**, a browser-based tool **(MaxDB Web database manager)**, and a command-line-based tool **(Max DB database manager CLI)**. The CLI version is used mainly to create automated administration processes (scripts).

MaxDB Web dabase manager and MaxDB database manager CLI (DBMCLI) are available for all operating systems supported by MaxDB. The Max DB database manager GUI (DBMGUI) requires a Windows platform, although it can manage database instances on any operating system platform supported.

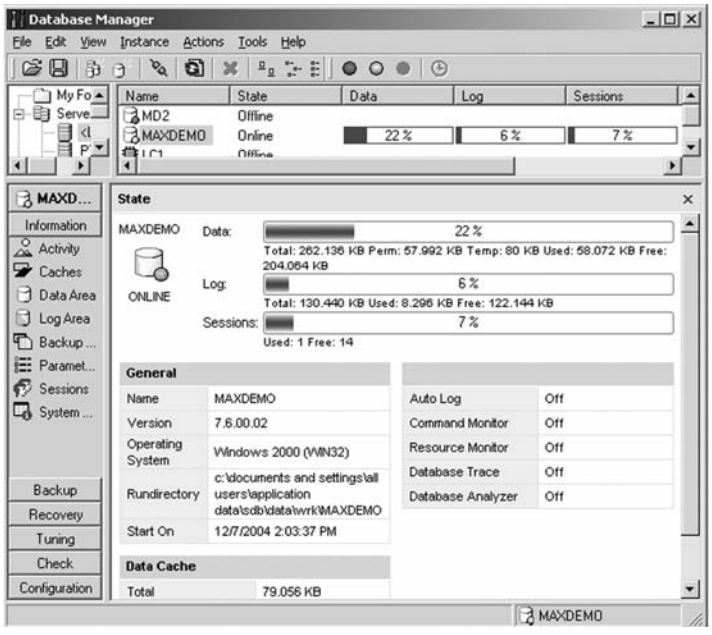


Figure 10: MaxDB Database Manager GUI

Figure 10 shows the overview status of a database instance (MAXDEMO). The screen is divided into areas so that you can navigate within the:

- System landscape (top left)
- Short overview of the status of the selected systems (top right)
- Pull-down menus for the various administration tasks (bottom left)
- Main window (central)

The main window shows detailed information about the action currently in focus.

Through MaxDB database manager, MaxDB provides the option of reacting to administrative database events by using programs, by automating the administration of database instances for routine cases (such as maintenance of optimizer statistics), or by using e-mail to inform people about an event that occurred (such as the shutdown of an instance).

MaxDB database manager is intended purely as a maintenance and monitoring tool for database operation. It does not grant any access to the data or to the definition of database objects.

Defining and Changing Data Using SQL

The **MaxDB SQL studio, MaxDB Web SQL,** and **MaxDB SQLCLI** query tools serve to:

- Access the data and database catalog of MaxDB instances using SQL statements
- Insert and process data
- Query data
- Create and manage database objects and database users

Two options are available: the Windows-based MaxDB SQL studio and the browser-based MaxDB Web SQL. MaxDB SQLCLI can be used to execute SQL commands on a command-line basis – for example, from script languages.

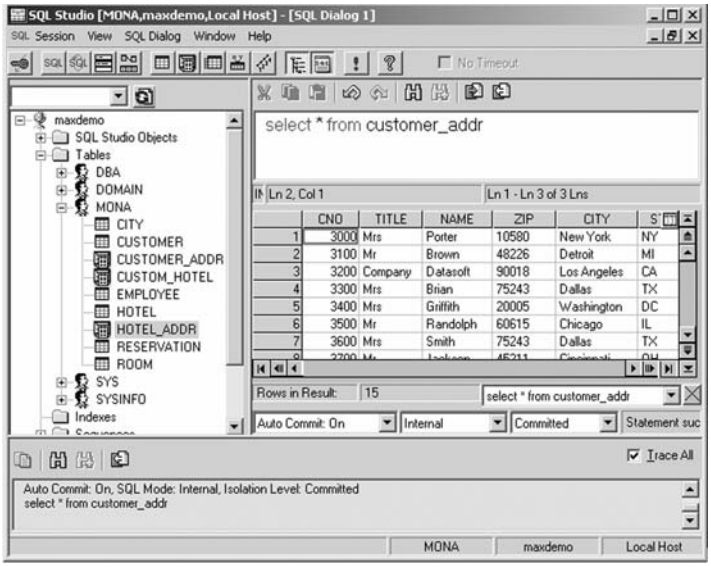


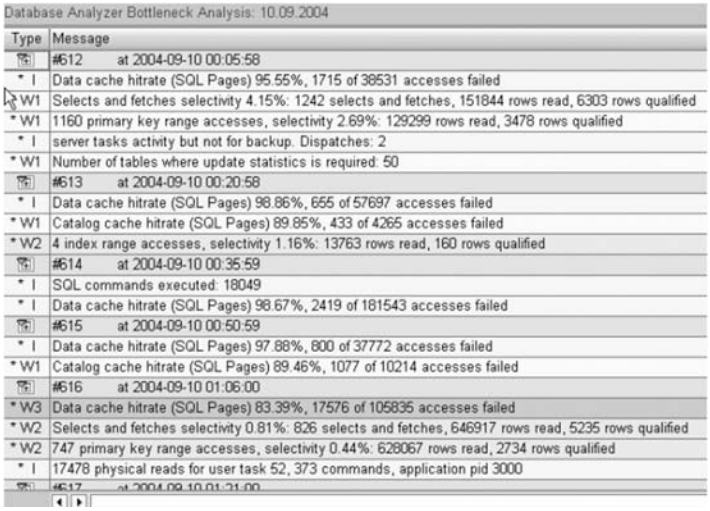
Figure 11: Querying with MaxDB SQL Studio

MaxDB SQL studio offers navigation support within the objects of the database catalog (Figure 11, left). The SQL dialog shown in Figure 11 is one among three dialog types permitting you to enter the SQL command directly in the SQL syntax (top right). The result is displayed in report format. In the lower area, a trace can be activated that logs the commands sent to the database together with the corresponding messages. The other query modes offer form-based and visual support in creating the database query and implicitly generate the required SQL syntax.

Monitoring the Database and Analyzing Problems

MaxDB database analyzer is an expert tool for MaxDB monitoring and tuning. It retrieves, logs, and assesses performance-relevant data at regular time intervals. The findings obtained are categorized in five classes: information, three warning levels, and error messages. As a result, you can hand the monitoring of database instances over to a tool, reducing the burden on the DBA expert.

MaxDB database analyzer facilitates the troubleshooting required in the event of performance or throughput problems with a MaxDB instance. It can be used to analyze the database configuration (caches, heap area, parameters), the synchronization (locks, critical sections), the optimization of database queries (processing strategy, indexes, optimizer statistics), or the hardware configuration. The analysis concludes with recommendations – for example, a suggestion to create additional indexes or to increase the size of the database cache.



Database Analyzer Bottleneck Analysis: 10.09.2004

Type	Message
	#612 at 2004-09-10 00:05:58
* I	Data cache hitrate (SQL Pages) 95.55%, 1715 of 38531 accesses failed
* W1	Selects and fetches selectivity 4.15%: 1242 selects and fetches, 151844 rows read, 6303 rows qualified
* W1	1160 primary key range accesses, selectivity 2.69%: 129299 rows read, 3478 rows qualified
* I	server tasks activity but not for backup. Dispatches: 2
* W1	Number of tables where update statistics is required: 50
	#613 at 2004-09-10 00:20:58
* I	Data cache hitrate (SQL Pages) 98.86%, 655 of 57697 accesses failed
* W1	Catalog cache hitrate (SQL Pages) 89.85%, 433 of 4265 accesses failed
* W2	4 index range accesses, selectivity 1.16%: 13763 rows read, 160 rows qualified
	#614 at 2004-09-10 00:35:59
* I	SQL commands executed: 18049
* I	Data cache hitrate (SQL Pages) 98.67%, 2419 of 181543 accesses failed
	#615 at 2004-09-10 00:50:59
* I	Data cache hitrate (SQL Pages) 97.88%, 800 of 37772 accesses failed
* W1	Catalog cache hitrate (SQL Pages) 89.46%, 1077 of 10214 accesses failed
	#616 at 2004-09-10 01:06:00
* W3	Data cache hitrate (SQL Pages) 83.39%, 17576 of 105835 accesses failed
* W2	Selects and fetches selectivity 0.81%: 826 selects and fetches, 646917 rows read, 5235 rows qualified
* W2	747 primary key range accesses, selectivity 0.44%: 628067 rows read, 2734 rows qualified
* I	17478 physical reads for user task 52, 373 commands, application pid 3000
	#617 at 2004-09-10 01:21:00

Figure 12: MaxDB Database Analyzer Log

Importing and Exporting Data

MaxDB loader lets you import application data from the file system to a MaxDB instance, or export data from a MaxDB instance to the file system. Because it supports a large number of data formats and record layouts, it can often replace a dedicated application program for loading or unloading data sets. And because of its close connection to the MaxDB instance, MaxDB loader usually offers performance advantages. It also supports the transport of any data from one MaxDB instance to another – one that may also run on different hardware architectures or different operating systems. The data can be individual tables, all tables belonging to a specific database user, or even the complete content of a database. MaxDB loader simply exports it from database A to a transport medium such as a file, a pipe, or a tape, and then imports from the transport medium into database B.

Replicating Data

MaxDB synchronization manager handles replicated (redundant) data in different MaxDB instances. Typically, it is used for mobile MaxDB databases on laptops that are occasionally synchronized with a central MaxDB instance. In this context, synchronizing means merging the changes that were made in the central instance or in the laptop instance.

To synchronize data, one MaxDB instance is defined as the master instance. Further, the landscape of the client instances is defined and the master database sections to be replicated are determined. After the replicated data has been initially transferred, all changes are logged in the master as well as in all client instances, and then merged during the next synchronization. With bidirectional replication, some "change" conflicts may have to be resolved manually.

The message server of the MaxDB synchronization manager is a dedicated MaxDB instance that is independent from the master and the client instances. It buffers all changes until the propagation for these changes has been completed. MaxDB synchronization manager thus decouples the availability of the master and the client instances from the time of synchronization.

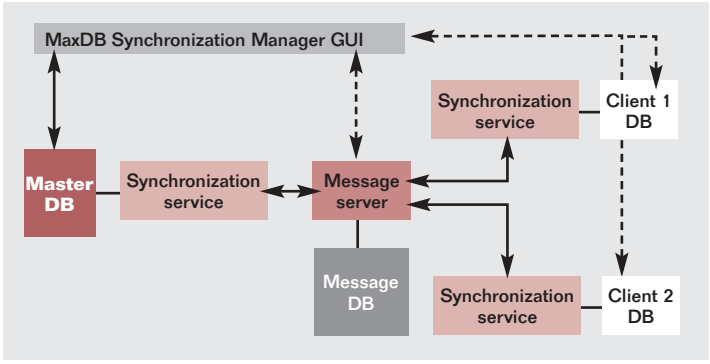


Figure 13: MaxDB Synchronization Manager Architecture

Operating a MaxDB instance on conventional laptops is no problem and does not require any administration. For smaller devices (for example, Pocket PC or BlackBerry) there is **MinDB**, a pure Java DBMS with minimal resource requirements that can be operated via a JDBC interface. Like MaxDB, MinDB can serve in bidirectional replication via the MaxDB synchronization manager.

MinDB provides a reduced SQL feature set but offers transactions, multiple sessions, and data persistence to external media such as memory sticks.

DEVELOPING DATABASE APPLICATIONS

MaxDB provides a series of standardized programming interfaces for developing SQL-based applications in different programming environments.

Programming Interfaces

For application development, MaxDB provides an open database connectivity (**ODBC**) driver in accordance with ODBC 3.5.

For the Java community, it provides a Java database connectivity (**JDBC**) driver according to JDBC 3.0. Additionally, MaxDB provides its native call-level interface SQL database connectivity (**SQLDBC**) on all supported operating system platforms – for example, for programming in C or C++. The architecture of SQLDBC is similar to that of ODBC and JDBC. SQLDBC supports an unlimited number of connections and SQL statements, is fully Unicode compliant, provides flexible result sets, has an additional large-object interface, and supports a wide range of SQL and C data types.

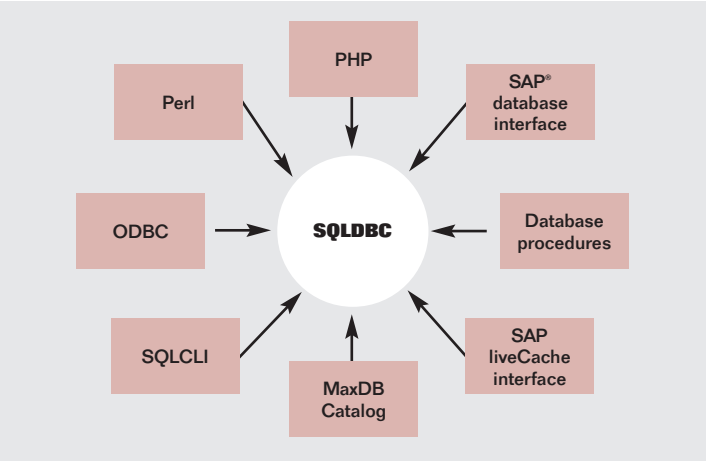


Figure 14: MaxDB SQLDBC Programming Interface

Apart from supporting the major development environments and programming languages via ODBC and JDBC, MaxDB provides native drivers for the most popular script languages – **PHP**, **Perl**, and **Python**, for example – used in open source projects.

IMPLEMENTING MaxDB

At the MySQL Web site (**www.mysql.com**) you can either download the source code of MaxDB and compile it yourself (creating an executable program tailored for your specific system environment) or you can use one of the precompiled and quality-assured executable programs available for the prominent operating system platforms. These precompiled "executables" are kept up to date and include the latest "patches." Product documentation and release information are also available on the Web site, along with a moderated mailing list that enables MaxDB users to support and educate each other and a Web-based problem tracking system that enables users to monitor the MaxDB problem-resolution process and check whether a problem has already been reported by other users.

Download

For SAP customers, MaxDB software is included in the delivery scope of their SAP solution. Further, it is available through the SAP Service Marketplace (**http://service.sap.com**).

To download the latest open source version of MaxDB software, including all tools and interfaces, go to **http://dev.mysql.com/downloads/maxdb**. This "all" software package contains everything required to create and work with MaxDB including the database kernel, the administration and access tools, and the programming interface manager. It does not, however, contain the MaxDB database manager GUI and MaxDB SQL studio tools, available only for Windows. You will find these tools at **http://dev.mysql.com/downloads/maxdb/clients_75.html**. The interfaces can also be downloaded as separate packages.

Downloads of the MaxDB development environment and released versions of the MaxDB sources are also available at **www.mysql.com/maxdb**. You can use CVS to download the latest MaxDB source code.

Former versions of MaxDB are archived at **http://downloads.mysql.com/archives.php?p=maxdb-7.5.00**.

Installation

The **MaxDB installation manager** installs MaxDB on all supported operating systems. During installation, you can select one of the preconfigured desktop or laptop installations – or perform a custom installation if, for example, MaxDB is to be used as a server database.

The sample database mentioned in the MaxDB tutorial can be loaded as an option.

The screens in Figures 15 and 16 show the MaxDB installation manager.

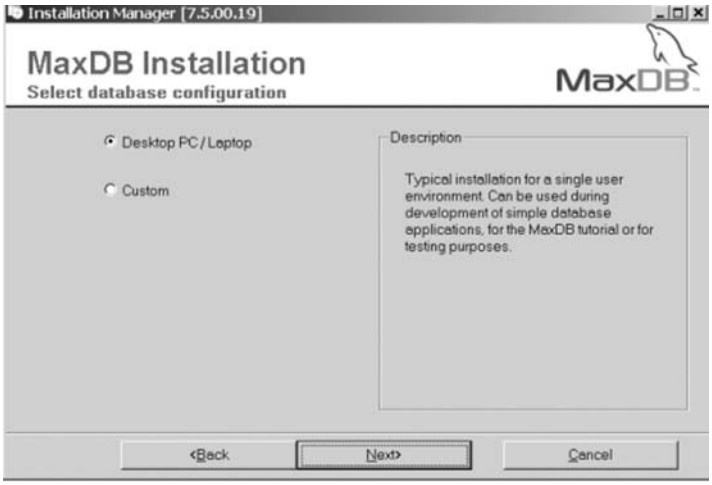

Installation Manager [7.5.00.19]
MaxDB Installation
Select database configuration
MaxDB
Desktop PC / Laptop
Custom
Description
Typical installation for a single user environment. Can be used during development of simple database applications, for the MaxDB tutorial or for testing purposes.
<Back
Next>
Cancel

Figure 15: MaxDB Installation Manager: Choosing a Database Template

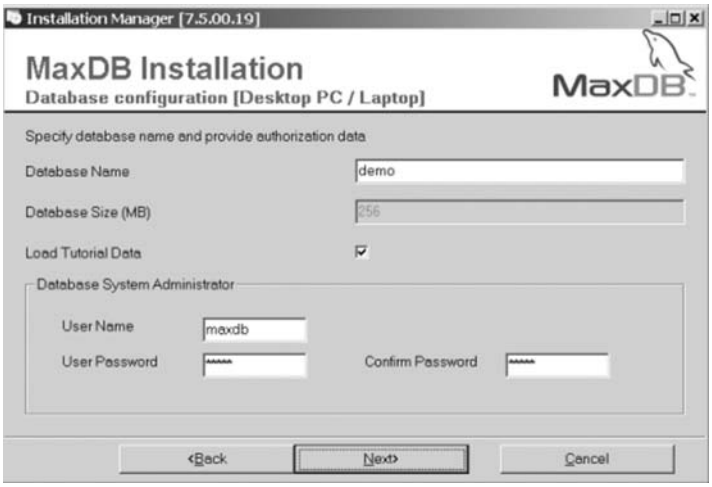


Figure 16: MaxDB Installation Manager: Configuring the Database

As an alternative to the MaxDB installation manager, you can create MaxDB instances using the MaxDB database manager in interactive mode or in a batch mode via a command-line script (see section on MaxDB database manager CLI).

Licensing Model

MaxDB is available in combination with SAP solutions or with the SAP NetWeaver technology and integration platform from SAP AG (**www.sap.com** or **www.sdn.sap.com**). Costs are based on the SAP price list. For a quotation on MaxDB for SAP applications, please go to **www.sap.com/contactsap**, where you will be directed to the nearest SAP sales office. MaxDB is also available as a stand-alone database system for your own or third-party solutions. Go to MySQL AB (**www.mysql.com/maxdb**).

MySQL AB offers MaxDB under its "dual licensing" model. In this model, users may choose to use MaxDB under the free software/open source GPL or under a commercial license.

MaxDB software is covered 100% by the GPL; thus, if your application or solution connecting to the database is 100% GPL compliant, you will be exempt from royalty. This is a great opportunity for the open source community, since it allows educational and trial usage and even supports the development phase of application software.

The commercial license is an agreement with MySQL AB for organizations that require MaxDB product support or do not wish to publish their application source code under GPL.

Consulting

If MaxDB has been supplied by SAP AG and is used with SAP solutions, consulting is provided by SAP AG.

MySQL AB also offers consulting services for MaxDB that cover the following areas:

- Migration of data and functionality from other DBMSs to MaxDB
- Adaptation of existing applications to MaxDB, including database configuration, tuning, and optimization
- Assistance from database experts on application development projects
- On-site training tailored to customer requirements

Support

If MaxDB is used with SAP solutions or with SAP NetWeaver, support is included in the SAP maintenance contract and is provided by SAP AG.

MySQL AB offers commercial MaxDB support to customers with a commercial license for MaxDB. Two support levels are available:

- Entry-level support, which comprises unlimited e-mail communication with MaxDB experts from MySQL AB
- Premium support, which comprises e-mail, login, and up to 24/7 phone support to customers with business-critical applications on MaxDB

For users of the GPL version, MySQL AB provides the MaxDB mailing list (at **http://lists.mysql.com/maxdb**), where MaxDB users can help each other. Many expert users and MaxDB developers with years of experience actively participate in the list. Participating in the mailing list is an excellent way for you to become a MaxDB expert. You can also use the list to communicate feature requests as well as problem reports to the MaxDB development team.

MaxDB developers handle change requests and problem resolutions in the internal MaxDB problem tracking system (PTS). The URL of the corresponding problem message is subsequently sent to the subscribers of the mailing list and can be used to monitor problem resolution via the publicly available Web PTS.

Training and Certification

Training options are available from both SAP and SQL.

SAP Training

SAP currently offers one MaxDB course and two workshops for SAP customers:

- **Database Administration MaxDB (ADM515)**
  Application developers, experienced administrators, and users working directly with MaxDB should attend this course. It is also recommended for IT managers who want to learn more about MaxDB as well as users of the previous versions of MaxDB (SAP DB and ADABAS D) who plan to upgrade to MaxDB.
  In this course, you will become familiar with the MaxDB database and its administration tools (MaxDB database manager GUI, MaxDB database manager CLI) as well as with the administration of SAP systems (CCMS) so that you can implement appropriate concepts for backing up MaxDB and discovering and analyzing performance bottlenecks.
- **Empowering Workshop MaxDB Performance Monitoring and Optimization (UMEW60)**
  This workshop deals with the basics of the MaxDB architecture. Here you will learn how the database influences general system performance and how database-intensive SAP transactions can be identified. You will also learn how to monitor MaxDB components such as the data cache, volumes, and locks. You will practice dealing with bottleneck analysis using the database analyzer and practice monitoring expensive SQL statements using the command monitor and the resource monitor. MaxDB experts will explain the concepts of the cost-based optimizer and the secondary index design. They will also teach you how to optimize expensive SQL statements and how to structure a performance analysis cookbook.
- **MaxDB Internals Workshop (WB550)**
  This workshop deals with the internals of MaxDB on a detailed technical level. Intended for expert-level users and administrators, it shows logging, locking, parameter handling, error diagnostics, storage handling, and other important internal algorithms and data structures for MaxDB.

Training by MySQL

MySQL AB offers two professional training courses for MaxDB users:

- **MaxDB Administration**
  This course focuses on using MaxDB from an administrator's point of view. You will learn to install a MaxDB instance, to understand the capabilities of the database in different scenarios, and to use the administration tools provided. After the course, you will be able to understand and handle advanced administrative and monitoring tasks on your MaxDB.
- **Using and Developing MaxDB**
  This course focuses on using MaxDB from a developer's point of view. You will learn in detail how to set up and use the database when programming an application as well as how to handle the MaxDB capabilities. After the course, you will be able to design your database according to your needs and know how to optimize it to achieve the best possible performance.

Combined with adequate practical experience, these courses will prepare you for the MaxDB 7.5 Administrator and Developer Certifications.

Certification by SAP

SAP offers certification to administer SAP systems in a 20-day basic training program, followed by a choice of database-dependent administration courses. The MaxDB course is a three-day course.

Having passed the examination after the course, you will be certified as "Technology Consultant SAP Netweaver/SAP WebAS on MaxDB."

Certification by MySQL

Certification for MaxDB is under way.

Technical Specifications

Description	Maximum Value
Database size	32 TB (with 8 KB page size)
Number of data volumes/size per volume	63 volumes/512 GB 127 volumes/256 GB ... 4,095 volumes/8 GB
Number of log volumes	32
Log size	32 TB
SQL statement length	Maximum 128 KB, default = 64 KB
Identifier length	32 characters
Numeric precision	38 digits
Number of tables	Unlimited
Number of columns per table (with key)	1,024
Number of columns per table (without key)	1,023
Number of primary key columns per table	512
Number of columns in a compound index	16
Number of foreign key columns per table	16
Number of columns in an "order" or a "group" clause	128
Number of columns in a "select" statement	1,023
Number of columns in an "insert" statement	1,024
Number of columns in a result table	1,023
Number of join tables in a "select" statement	64
Number of triggers per base table	3
Number of indexes per table	255
Number of referring "constraint" definitions (foreign key dependencies) per table	unlimited
Number of references per table	unlimited
Number of rows per table	limited by database size
Internal length of a table row	8,088 bytes
Total of internal lengths of all primary key columns	1,024 bytes
Total of internal lengths of all foreign key columns	1,024 bytes
Total of internal lengths of all columns belonging to an index	1,024 bytes
Internal length of a BLOB, CLOB, or LONG column (Note: BLOB is a binary large object, CLOB is a character large object, LONG is a MaxDB large object of any type)	2 GB
Length of columns in an "order" or a "group" clause	1,020 bytes
Nested trigger levels	Unlimited
Nested subqueries	127
Number of join conditions in a "where" clause of a "select" statement	128
Number of correlated columns in an SQL statement	64
Number of correlated tables in an SQL statement	16
Number of parameters in an SQL statement	2,000

MaxDB Documentation Overview

The following table contains an overview of the entire user documentation available for MaxDB:

Installation	
Database Software Installation Guide	Standard installation and "uninstallation" of the database software
Installation Manual	Installation, update, and "uninstallation" of installation profiles and individual software components for the database; update of database instances, including software
SAP Web Application Server Installation on UNIX: MySQL MaxDB	Installation of an SAP system on UNIX, only available for SAP customers on the SAP Service Marketplace
SAP Web Application Server Installation on Microsoft Windows: MySQL MaxDB	Installation of an SAP system on Microsoft Windows, only available for SAP customers on the SAP Service Marketplace
SAP Web Application Server: Homogeneous and Heterogeneous System Copy	Copying SAP systems, only available for SAP customers on the SAP Service Marketplace

Basic Knowledge	
Concepts of the Database System	Architecture of the database system, database parameters, database tools, users, and security concepts, logging, B* trees, standby databases, and more
Reference Manual	SQL statements and their syntax, system tables, special features in the MaxDB “Oracle” mode
SQL Tutorial	Use of SQL language for operations with data and tables of the MySQL MaxDB system using the demo schema HOTELDB as an example
Messages	List of all messages
Glossary	Database system terms
SAP Security Guide: MySQL MaxDB	Recommended security actions for the database system
High Availability for the MySQL MaxDB Database	Conditions for ensuring high availability of a MaxDB instance in an SAP solution
User manual: SAP liveCache	Notes on managing SAP liveCache

Tools	
Database Manager Database Manager GUI Web DBM Database Manager CLI	Creating and managing database instances Windows tool Browser tool Command-line tool
Database Analyzer	Monitoring performance and resources of database instances, detecting pain points
Loader	Loading and unloading data into and from MaxDB instances
Query Tools SQL Studio Web SQL SQLCLI	SQL queries and catalog management Windows tool Browser tool Command-line tool
WebDAV GUI	Administration of a MaxDB WebDAV repository for XML documents
Synchronization Manager	Administration of a MaxDB landscape with replicated datasets

Interfaces	
ODBC Manual	MaxDB ODBC driver for Windows developers
JDBC Manual	MaxDB JDBC driver for Java developers
SQLDBC Manual	Native MaxDB driver for all operating system environments, especially for C and C++
PHP Manual	MaxDB PHP driver for PHP developers
Perl Manual	MaxDB Perl driver for Perl developers
Python Manual	MaxDB Python driver for Python developers
Backint for MaxDB	Description of the MaxDB Backint interface for external backup tools

Development	
Development Environment	Use of the MaxDB open source development environment
Web-Based Problem Tracking System	Browser-based system for tracking problem messages and processing

Disclaimer: This document reflects current product layout and current planning. Contents may be changed without prior notice and are in no way binding upon SAP AG or MySQL AB.

WHY YOU NEED MaxDB

MaxDB provides the solutions and tools that will help run your business efficiently in today's challenging enterprise environment. It features:

- Proven DBMS technology
- Attractive pricing
- Easy configuration and administration
- Around-the-clock operation
- Elaborate backup and restore capabilities

Not only does MaxDB support a large number of users and demanding workloads, it is available for all major operating system platforms. Best of all, it is bundled with the SAP NetWeaver platform and supports all major SAP solutions.

For More Information

To find out more about MaxDB and learn how the database can transform your enterprise management capabilities, go to **www.mysql.com/maxdb** or **www.sdn.sap.com > Developer Areas > Technologies > MaxDB & SAP liveCache** – or call your local SAP representative.

www.sap.com/contactsap

50 074 399 (05/05)

THE BEST-RUN BUSINESSES RUN SAP™